

PHFR 000039  
LIS



Europäisches  
Patentamt

European  
Patent Office

Office européen  
des brevets



Bescheinigung

Certificate

Attestation

Die angehefteten Unterla-  
gen stimmen mit der  
ursprünglich eingereichten  
Fassung der auf dem näch-  
sten Blatt bezeichneten  
europäischen Patentanmel-  
dung überein.

The attached documents  
are exact copies of the  
European patent application  
described on the following  
page, as originally filed.

Les documents fixés à  
cette attestation sont  
conformes à la version  
initialement déposée de  
la demande de brevet  
européen spécifiée à la  
page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

00401075.7

Der Präsident des Europäischen Patentamts;  
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets  
p.o.

I.L.C. HATTEN-HECKMAN

DEN HAAG, DEN  
THE HAGUE, 01/12/00  
LA HAYE, LE

**THIS PAGE BLANK (USPTO)**



Europäisches  
Patentamt

European  
Patent Office

Office européen  
des brevets

**Blatt 2 der Bescheinigung**  
**Sheet 2 of the certificate**  
**Page 2 de l'attestation**

Anmeldung Nr.:  
Application no.: 00401075.7  
Demande n°:

Anmeldetag:  
Date of filing: 18/04/00  
Date de dépôt:

Anmelder:  
Applicant(s):  
Demandeur(s):  
Koninklijke Philips Electronics N.V.  
5621 BA Eindhoven  
NETHERLANDS

Bezeichnung der Erfindung:  
Title of the invention:  
Titre de l'invention:  
Pixel-based data modifying method and device

In Anspruch genommene Priorität(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat:  
State:  
Pays:

Tag:  
Date:  
Date:

Aktenzeichen:  
File no.  
Numéro de dépôt:

Internationale Patentklassifikation:  
International Patent classification:  
Classification internationale des brevets:

/

Am Anmeldetag benannte Vertragsstaaten:  
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE/UK  
Etats contractants désignés lors du dépôt:

Bemerkungen:  
Remarks:  
Remarques:

**THIS PAGE BLANK (USPTO)**

**"Pixel-based data modifying method and device"****FIELD OF THE INVENTION**

- 5 The present invention relates to a method of modifying data in an encoded data signal, comprising at least :
- a decoding step for decoding said encoded data signal, and providing a decoded data signal,
  - a re-encoding step performed on a modified data signal and generating a coding error,
  - 10 - a prediction step for providing a motion-compensated signal from said coding error, and comprising at least a subtracting sub-step between an input data signal obtained at least from said decoded data signal, and said motion-compensated signal for carrying out said modified data signal,

15 The invention also relates to video processing devices for carrying out said method. This invention, for instance, may be used when a broadcaster wants to introduce additional data into a sequence of coded pictures. This invention finds applications not only in the field of MPEG-2 compression, but more generally in any digital video data compression system.

**BACKGROUND OF THE INVENTION**

20 Modifying data in an encoded data signal has become a vital functionality in studio-editing environment. A possible solution has been proposed in the international patent application WO 99/51033 (PHF98546). This patent application describes a method and its corresponding device for modifying data in an encoded data signal. This method allows an additional data signal insertion, e.g. a logo inserting, in an MPEG-2 bitstream thanks to bit rate

25 transcoding. Logo insertion comes as an extension of the bit rate transcoder. The corresponding scheme, depicted in Fig.1, comprises a transcoding module 101 and a logo addition branch 102. The general outline of the transcoding module 101, well known to a person skilled in the art, comprises :

- a residue decoding branch 118 for receiving the input signal 125 and providing a decoded data signal Error\_I(n). This branch comprises in series a variable length decoding 107, an inverse quantization 108 followed by an inverse discrete cosine transform 109.
- a re-encoding/decoding branch 120 for respectively providing the output signal 126 and its decoded version. The re-encoding part, for providing said output signal, comprises in series a discrete cosine transform 110, a quantization 111, a variable length coding 112
- 35 followed by a buffer 113, and regulation means 114 ensuring a constant picture quality of the output signal 126, and a first subtracter 122 generating a coding error. The decoding part comprises in series an inverse quantization 115 followed by an inverse discrete cosine transform 116.

- an intermediary branch 119 comprising a motion compensation 105 using motion vectors  $V(n)$  of the input signal, its associated memory 106 storing a previous signal, and a second subtracter 123. This branch, also called prediction loop, avoids the quality drift in the output signal by applying a motion compensation to said coding error generated during the re-encoding step.

The logo addition branch 102 is implemented thanks to a residue addition to the decoded signal  $Error\_I(n)$ , by means of the adding sub-step 121. This residue is formed by subtracting an additional data signal  $Logo(n)$  referred as 127 with a motion-compensated logo prediction referred as 129, obtained by means of the motion compensation sub-step 103, that is based on reference pictures containing logo previously stored in memory 104 and that uses the same vectors  $V(n)$  as the main input signal.

In the scheme relative to the prior art depicted in Fig.1, two motion compensations are performed : a first one 105, well known and provided for correcting the quality drift on P and B pictures introduced by the quantization sub-step 111, and a second one 103 on additional data signal 127. This motion compensation 103 generates said motion compensated signal  $PRED( Logo(n-1), V(n) )$  referred as 129 that is subtracted to said signal 127. Said motion-compensated signal is indeed essential since it cancels in the input signal of the re-encoding step, undesired parts of the signal relative to signal 127 previously motion-compensated by 105. Moreover, as a motion compensation always requires a storage of a previous signal, two memories 104 and 106 are also needed. Then, with these two motion compensation operations and two memory blocks, this solution remains complex not only in terms CPU burden, but also expensive in terms of memory storage.

## **SUMMARY OF THE INVENTION**

It is an object of the invention to provide a method of modifying data in an encoded data signal which requires less memory capacity and central processing units (CPU) burden.

The method of modifying data according to the invention is characterized in that it comprises :

- a first sub-step for adding an additional data signal to said decoded data signal, for providing said input data signal,
- a second sub-step for adding said additional data signal to said coding error, said motion-compensated signal resulting from the motion compensation of said second adding sub-step output signal.

A variant of the previously characterized method is also proposed. It is characterized in that it comprises a sub-step for adding an additional data signal to said modified data signal, before said re-encoding step.

5 The corresponding schemes, respectively depicted in Fig.2 and Fig.3, are based on data addition in the pixel domain of the additional data signal 127 with the decoded data signal relative to the input data signal, or with signal situated in the transcoder drift correction loop, by means of said adding sub-steps.

10 According to the invention, compared to the prior art solution, no more separate motion compensation is applied to logo data since the motion compensation relative to logo data is merged with the motion compensation used to correct the drift correction in the transcoder loop. By this way, the invention comprises a minimum number of functional sub-steps leading to a cost-effective solution. Indeed, only one set of motion compensation and its associated memory storage is used, such simplifications being possible in taking advantage of combinations between different sub-steps, and using their own characteristics as the linearity of the motion compensation.

15

Another object of the invention is also to propose devices for implementing the above-mentioned schemes.

To this end, the invention relates in a first implementation, to a transcoding device for adding data to an encoded data signal, comprising :

- 20 - a first means for adding an additional data signal to said decoded data signal, for providing said input data signal,
- a second means for adding said additional data signal to said coding error, said motion-compensated signal resulting from the motion compensation of said second means output signal.

25

The invention also relates in a second implementation, to a transcoding device for adding data to an encoded data signal, characterized in that it comprises means for adding an additional data signal to said modified data signal, before re-encoding means.

30

Detailed explanations and other aspects of the invention are hereinafter given.

### **BRIEF DESCRIPTION OF THE DRAWINGS**

35 The particular aspects of the invention will now be explained with reference to the embodiments described hereinafter and considered in connection with the accompanying drawings, in which identical parts or sub-steps are designated in the same manner :

Fig.1 illustrates the outline, as known in the prior art, of a transcoder with its logo insertion branch,

Fig.2 illustrates a first embodiment of the technical solution according to the present invention.

Fig.3 illustrates a second embodiment of the technical solution according to the present invention.

### **DETAILED DESCRIPTION OF THE INVENTION**

5 As previously said, the present invention aims at reducing the cost of the prior art method for modifying data in an encoded data signal. Such invention is well adapted in the case of MPEG-2 coded video signals as input signal, but it will be apparent to a person skilled in the art that such a method is applicable to any coded signal that has been encoded with a block-based compression method such as for example, the one described in MPEG-4, H.261 or H.263  
10 standards. In the following, the invention will be detailed assuming that encoded video signals are compliant with the MPEG-2 international video standard (Moving Pictures Experts Group, ISO/IEC 13818-2).

15 Fig.2 depicts the first cost-effective scheme of data insertion in an encoded data signal, object of the present invention. This scheme re-uses the scheme of the transcoder described just above, on which are inserted sub-steps aiming at modifying the input signal. Indeed, the input signal is modified by pixel-based data signal 127 simultaneously introduced thanks to two adding sub-steps :

- 20 • the sub-step 121 placed at the output of the error residue decoding, more precisely on the output signal of the inverse discrete cosine transform 109. The modification of the input signal, e.g. in case of a logo insertion, is therefore first implemented by means of an addition between the inserted data signal 127 and the incoming data signal Error\_I(n). This addition results in a signal corresponding to the positive input of the subtracting sub-step 123.
- 25 • the sub-step 124 at the input of the memory 206 relative to the motion compensation 205. The modification of the input signal, e.g. in case of a logo addition, is therefore secondly implemented by means of an addition between the inserted data signal 127 and the output signal of the subtracting sub-step 122. This addition results in a signal corresponding to the input signal of the memory 206.

30

From an algorithmic point of view, this first scheme proposed according to the invention is equivalent to the one described in the prior art of Fig.1, as it can be recursively demonstrated hereinafter.

35 Let note the following notations relative to the demonstration :

- $V(n)$  : vectors of picture number  $n$ ,
- $I(n)$  : decoded input picture number  $n$ ,
- $Error\_I(n)$  : error residue of input picture number  $n$ ,



- $O1(n)$  : decoded picture number  $n$  corresponding to the output signal of Fig.1,
- $MEM1(n)$  : picture number  $n$  stored in the frame memory 106,
- $O2(n)$  : decoded picture number  $n$  corresponding to the output signal of Fig.2,
- $MEM2(n)$  : picture number  $n$  stored in the frame memory 206,
- 5 •  $PRED(X(n), V(n+1))$  : motion compensation of picture  $X(n)$  in using vectors  $V(n+1)$ . It corresponds to a predicted version of picture  $X(n+1)$ ,
- $T$  : transform defined by  $T(x) = IDCT( IQ( Q( DCT(x) ) ) )$ .

Note that the decoded pictures  $I(n)$ ,  $O1(n)$  and  $O2(n)$  are not represented on any figures since only compressed signals are accessible.

10

The equivalence between the prior art scheme and the scheme of Fig.2 will be demonstrated if for each  $n$ , the three following relations are valid :

(1)  $O1(n) = O2(n)$

(2)  $MEM1(n) = O1(n) - I(n) - Logo(n)$

15 (3)  $MEM2(n) = O2(n) - I(n)$

Of course, in this demonstration, the input signal and the inserted data signal 127 of Fig.1 and Fig.2 are supposed to be identical.

20 For the case where  $n = 0$  that corresponds to an Intra-coded picture, it can be written :

(4)  $Error\_I(0) = I(0)$

From Fig.1, it can be deduced :

(5)  $O1(0) = T( I(0) + Logo(0) )$

25 (6)  $MEM1(0) = O1(0) - I(0) - Logo(0)$

From Fig.2, it can be deduced :

(7)  $O2(0) = T( I(0) + Logo(0) )$

(8)  $MEM2(0) = O2(0) - I(0)$

30

From relations (5), (6), (7) and (8), it is possible to conclude that relations (1), (2) and (3) are valid for  $n = 0$ . Let suppose they are still valid at the rank  $n$ , and let demonstrate that (1), (2) and (3) are also valid at the rank  $(n+1)$ .

35 Let now introduce the terms  $A(n+1)$  and  $B(n+1)$  as :

$$A(n+1) = Error\_I(n+1) + Logo(n+1) - PRED( Logo(n) , V(n+1) ) - PRED( MEM1(n), V(n+1) )$$

$$B(n+1) = Error\_I(n+1) + Logo(n+1) - PRED( MEM2(n), V(n+1) )$$

Since (2) and (3) are valid at the rank  $n$ , and since the motion compensation is linear,  $A(n+1)$  and  $B(n+1)$  becomes :

$$(9) \quad A(n+1) = \text{Error\_I}(n+1) + \text{PRED}(I(n), V(n+1)) + \text{Logo}(n+1) - \text{PRED}(O1(n), V(n+1))$$

$$(10) \quad A(n+1) = I(n+1) + \text{Logo}(n+1) - \text{PRED}(O1(n), V(n+1))$$

$$5 \quad (11) \quad B(n+1) = \text{Error\_I}(n+1) + \text{PRED}(I(n), V(n+1)) + \text{Logo}(n+1) - \text{PRED}(O2(n), V(n+1))$$

$$(12) \quad B(n+1) = I(n+1) + \text{Logo}(n+1) - \text{PRED}(O2(n), V(n+1))$$

Since (1) is valid for at the rank  $n$ , relations (10) and (12) become :

$$(13) \quad A(n+1) = B(n+1) = I(n+1) + \text{Logo}(n+1) - \text{PRED}(O1(n), V(n+1))$$

10

From Fig.1 and Fig.2, it can be deduced :

$$(14) \quad O1(n+1) = T(A(n+1)) + \text{PRED}(O1(n), V(n+1))$$

$$(15) \quad \text{MEM1}(n+1) = T(A(n+1)) - A(n+1)$$

$$(16) \quad O2(n+1) = T(B(n+1)) + \text{PRED}(O2(n), V(n+1))$$

$$15 \quad (17) \quad \text{MEM2}(n+1) = T(B(n+1)) - B(n+1) + \text{Logo}(n+1)$$

From relation (13), (15) and (17), one can conclude that :

$$(18) \quad \text{MEM1}(n+1) = O1(n+1) - I(n+1) - \text{Logo}(n+1)$$

$$(19) \quad \text{MEM2}(n+1) = O2(n+1) - I(n+1)$$

20

Then relations (1), (2) and (3) are valid for the rank  $n+1$ , which proves the algorithmic equivalence between the scheme Fig.1 of the prior art, and the first proposed scheme depicted in Fig.2 and object of the invention. By this way, this proposed scheme ensures that the modified output signal has the same quality as the one of the prior art, but obtained in a more cost effective manner. Indeed, no more separate motion compensation and its associated memory on inserted data signal 127 is needed, since said data can be directly inserted in the transcoding pseudo-prediction loop, this simplification being justified by the linearity of the motion compensation. This merge of the two motion compensations - if the insertion of the two adding sub-steps 121 and 124, cost-less for most of digital signal processors, is excluded - represents a substantial gain in terms of CPU consumption and also memory storage.

25

30

The Fig.3 depicts another variant of the present invention. It is also based on a transcoder scheme identical as the one previously described and depicted in Fig.1. Compared to a transcoder architecture such as the one depicted in Fig.1, only a few modifications are done to obtain a change in the input signal. Indeed, the input signal is modified by signal 127 introduced by means of only one adding sub-step 121 placed at the input of the re-encoding step, more precisely on the input signal of discrete cosine transform 110. The modification of the input signal, e.g. in case of a logo addition, is therefore implemented by means of an addition between the inserted data signal 127 and the output signal of the subtracting sub-step

35

123. This addition results in a signal corresponding to the input of the discrete cosinus transform 110. From an algorithmic point of view, this second scheme is also equivalent to the prior art scheme of Fig.1.

5 Let note the following notations relative to the demonstration :

- **O3(n)** : decoded picture number n corresponding to the output signal of Fig.3,
- **MEM3(n)** : picture number n stored in the frame memory 206.

Note that the decoded pictures O3(n) is not represented on any figures since only compressed signals are accessible.

10

The same recursive demonstration can be done in proving for each n, the three following equations :

$$(20) \quad O1(n) = O3(n)$$

$$(2) \quad MEM1(n) = O1(n) - I(n) - Logo(n) \quad (\text{ever demonstrated just above})$$

15

$$(21) \quad MEM3(n) = O3(n) - I(n)$$

Of course, in this demonstration, the input signal and the inserted data signal 127 of Fig.1 and Fig.3 are supposed to be identical.

20 For the case where  $n = 0$  that corresponds to an Intra-coded picture, it can be written :

$$(22) \quad Error\_I(0) = I(0)$$

From Fig.3, it can be deduced :

$$(23) \quad O3(0) = T( I(0) + Logo(0) )$$

25

$$(24) \quad MEM3(0) = O3(0) - I(0)$$

From relations (5), (6), (23) and (24), it is possible to conclude that relations (20) and (21) are valid for  $n = 0$ . Let suppose they are still valid at the rank n, and let demonstrate that (20) and (21) are also valid at the rank (n+1).

30

Let now introduce the terms  $C(n+1)$  as :

$$C(n+1) = Error\_I(n+1) + Logo(n+1) - PRED( MEM3(n), V(n+1) )$$

Since (21) are valid at the rank n, and since the motion compensation is linear,  $C(n+1)$  becomes :

35

$$(25) \quad C(n+1) = Error\_I(n+1) + PRED( I(n), V(n+1) ) + Logo(n+1) - PRED( O3(n), V(n+1) )$$

$$(26) \quad C(n+1) = I(n+1) + Logo(n+1) - PRED( O3(n), V(n+1) )$$

Since (2) is valid for at the rank n, relation (26) becomes :

$$(27) \quad A(n+1) = C(n+1) = I(n+1) + Logo(n+1) - PRED( O1(n), V(n+1) )$$

From Fig.3, it can be deduced :

$$(28) \quad O3(n+1) = T( C(n+1) ) + \text{PRED}( O3(n), V(n+1) )$$

$$(29) \quad \text{MEM3}(n+1) = T( C(n+1) ) - C(n+1) + \text{Logo}(n+1)$$

From relation (15), (27) and (29), it can be can conclude that :

5     **(30)     MEM3(n+1) = O3(n+1) - I(n+1)**

Then relations (20) and (21) are valid for the rank  $n+1$ . Thus, this proposed scheme ensures that identical results will be obtained in the output signals of Fig.1 and Fig.3. No more separate motion compensation and its associated memory on signal 127 are needed, and said pixel-based data signal 127 is introduced thanks to the only cost-less adding sub-step ADD. In terms of CPU consumption and memory storage, this solution is nearly the same compared to the one of an isolated transcoder without data insertion, which is remarkable.

10     In Fig.2 and Fig.3 described above according to the invention, the input signal data is modified thanks to the insertion of the pixel-based data signal 127, by means of adding sub-steps. These inserted data may correspond to a logo, i.e. a single small picture, or a ticker, i.e. successive small different pictures. In both cases, each picture must be pixel-based, e.g. in being encoded according to the so-called bitmap format which corresponds to a rough digital image coding. Of course, before insertion by means of the adding sub-steps, said signal 127 may derive from an adapted pixel-based signal Logo\_ori(n) referred as 328 in order to optimize the quality of the output signal, as it is only represented on Fig.3 with step 317, for example in changing the luminance or the chrominance levels, as far as their format is still compatible. It is obvious that such adaptation does not restrict the scope and the pertinence of the present invention.

25     This method of modifying data in an encoded data signal can be implemented in several manners, such as by means of wired electronic circuits or, alternatively, by means of a set of instructions stored in a computer-readable medium, said instructions replacing at least a part of said circuits and being executable under the control of a computer or a digital processor in order to carry out the same functions as fulfilled in said replaced circuits. The invention then also relates to a computer-readable medium comprising a software module that includes computer executable instructions for performing the steps, or some steps, of the above described method.

**CLAIMS**

1. A method of modifying data in an encoded data signal comprising :

- a) a decoding step for decoding said encoded data signal, and providing a decoded data signal,
- 5 b) a re-encoding step performed on a modified data signal and generating a coding error,
- c) a prediction step for providing a motion-compensated signal from said coding error, and comprising at least a subtracting sub-step between an input data signal obtained at least from said decoded data signal, and said motion-compensated
- 10 signal for carrying out said modified data signal,

**characterized in that** it comprises :

- d) a first sub-step for adding an additional data signal to said decoded data signal, for providing said input data signal,
- 15 e) a second sub-step for adding said additional data signal to said coding error, said motion-compensated signal resulting from the motion compensation of said second adding sub-step output signal.

2. A method of modifying data in an encoded data signal comprising :

- a) a decoding step for decoding said encoded data signal, and providing a decoded data signal,
- 20 b) a re-encoding step performed on a modified data signal and generating a coding error,
- c) a prediction step for providing a motion-compensated signal from said coding error, and comprising at least a subtracting sub-step between an input data signal obtained at least from said decoded data signal, and said motion-compensated
- 25 signal for carrying out said modified data signal,

**characterized in that** it comprises a sub-step for adding an additional data signal to said modified data signal, before said re-encoding step.

30 3. A transcoding device for adding data to an encoded data signal, comprising :

- a) decoding means for decoding said encoded data signal, and providing a decoded data signal,
- b) re-encoding means performed on a modified data signal and generating a coding error,
- 35 c) prediction means for providing a motion-compensated signal from said coding error, and comprising at least subtracting means between an input data signal obtained at least from said decoded data signal, and said motion-compensated signal for carrying out said modified data signal,

**characterized in that** it comprises :

- d) a first means for adding an additional data signal to said decoded data signal, for providing said input data signal,
- e) a second means for adding said additional data signal to said coding error, said motion-compensated signal resulting from the motion compensation of said second means output signal.

5

4. A transcoding device for adding data to an encoded data signal, comprising :

- a) decoding means for decoding said encoded data signal, and providing a decoded data signal,
- b) re-encoding means performed on a modified data signal and generating a coding error,
- c) prediction means for providing a motion-compensated signal from said coding error, and comprising at least subtracting means between an input data signal obtained at least from said decoded data signal, and said motion-compensated signal for carrying out said modified data signal,

10

15

**characterized in that** it comprises means for adding an additional data signal to said modified data signal, before re-encoding means.

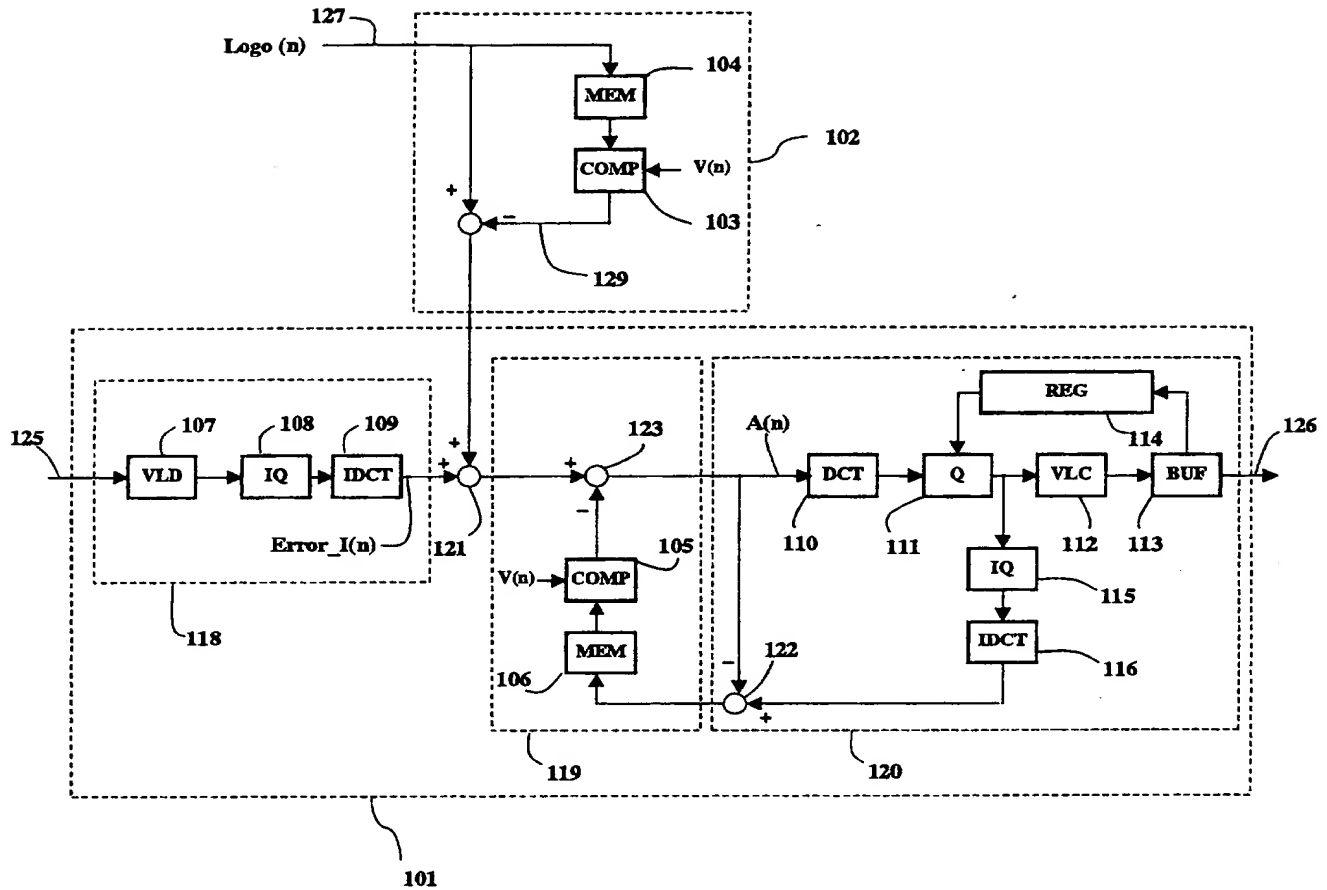
5. Computer program product, for a transcoding device for adding data to an encoded data signal, that comprises a set of instructions, which, when loaded into said device, causes the said device to carry out the method as claimed in claim 1.

20

6. Computer program product, for a transcoding device for adding data to an encoded data signal, that comprises a set of instructions, which, when loaded into said device, causes the said device to carry out the method as claimed in claim 2.

25

1 / 3



**FIG.1**  
PRIOR ART

2 / 3

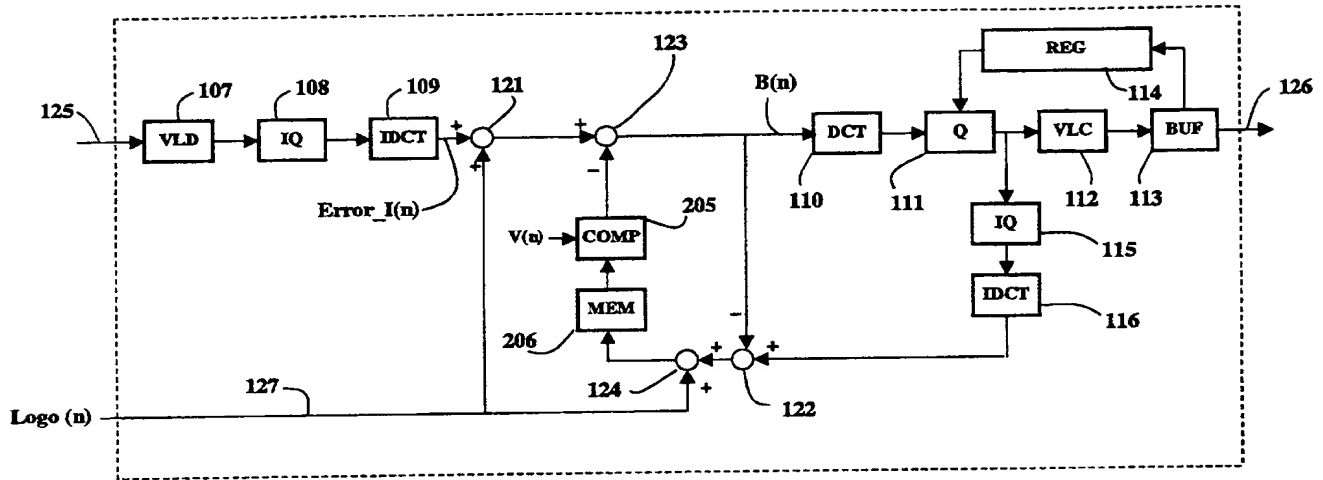
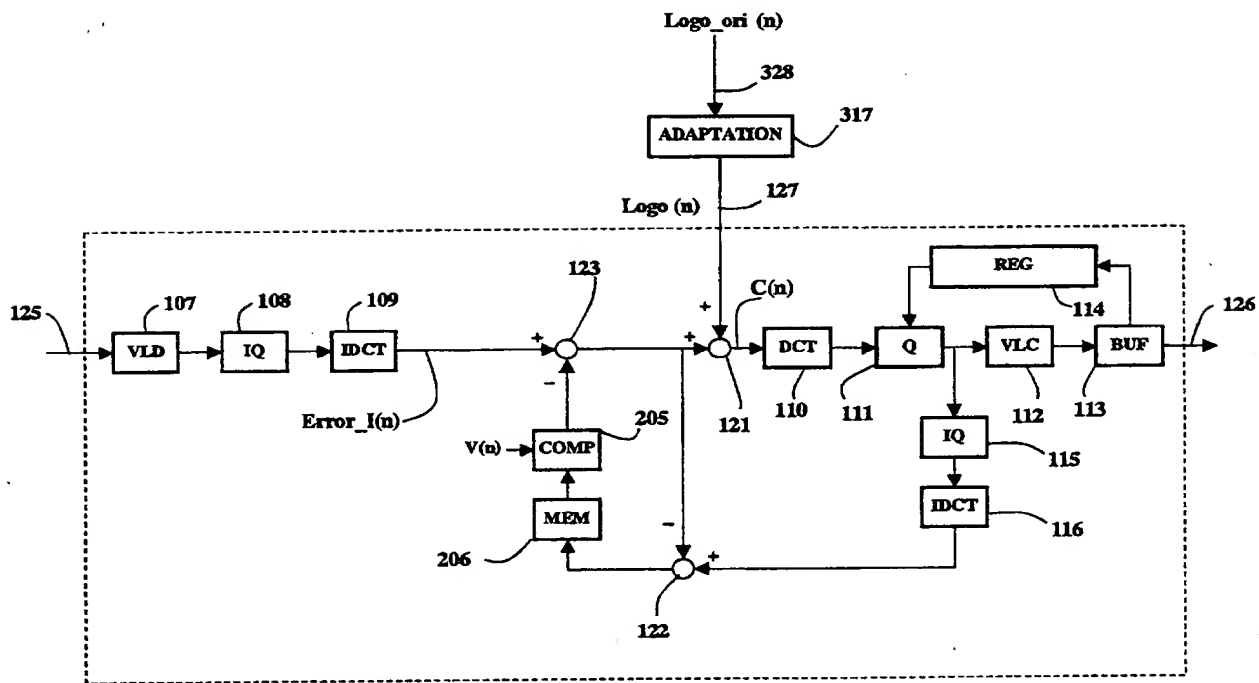


FIG.2





**FIG.3**

**THIS PAGE BLANK (USPTO)**